(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 October 2003 (23.10.2003)

PCT

(10) International Publication Number
WO 2003/088035 A3

(51) International Patent Classification[7]: G06F 9/44, 17/30

(21) International Application Number:
PCT/IB2003/001013

(22) International Filing Date: 19 March 2003 (19.03.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
02076460.1 15 April 2002 (15.04.2002) EP

(71) Applicant *(for all designated States except US)*: KONINKLIJKE PHILIPS ELECTRONICS N.V. [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: MORRIS, Steven [GB/—]; c/o Prof . Holstlaan 6, NL-5656 AA Eindhoven (NL).

(74) Agent: GROENENDAAL, Antonius, W., M.; Philips Intellectual Property & Standards, Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

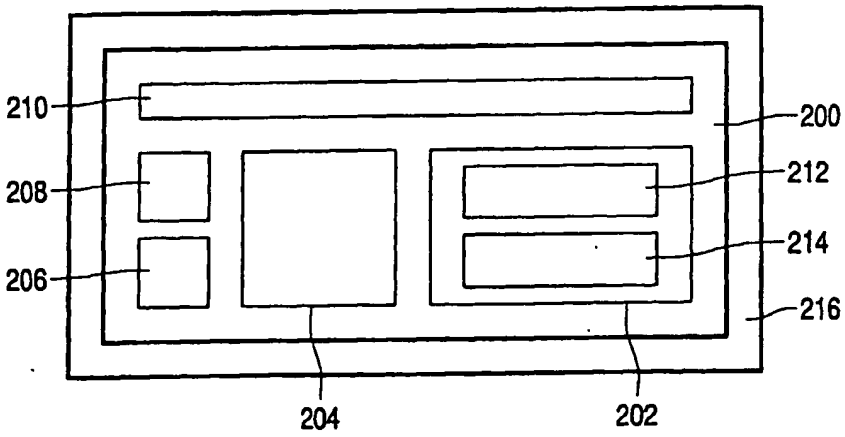
(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*
— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

(88) Date of publication of the international search report:
4 November 2004

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD, SYSTEM, COMPUTER PROGRAM PRODUCT AND STORAGE DEVICE FOR DISPLAYING A DOCUMENT



(57) Abstract: Handheld display devices have often limited display capabilities. In order to view a full HTML page (200), a method is provided that splits the original page into less complex units (210, 206, 208, 204, 212, 214, 202) so that they can be best displayed on these small devices. The document is parsed into a tree structure and the complexity of the tree is calculated. Depending on the complexity it is decided which part of the document must be displayed onto a new page.

WO 2003/088035 A3

INTERNATIONAL SEARCH REPORT

National Application No
T/IB 03/01013

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 G06F9/44 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, IBM-TDB, INSPEC, COMPENDEX

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	HORI M ET AL: "Annotation-based Web content transcoding" COMPUTER NETWORKS, ELSEVIER SCIENCE PUBLISHERS B.V., AMSTERDAM, NL, vol. 33, no. 1-6, June 2000 (2000-06), pages 197-211, XP004304767 ISSN: 1389-1286 the whole document	1-7
X	BICKMORE T ET AL: "WEB PAGE FILTERING AND RE-AUTHORING FOR MOBILE USERS" COMPUTER JOURNAL, OXFORD UNIVERSITY PRESS, SURREY, GB, vol. 42, no. 6, 1999, pages 534-546, XP000920338 ISSN: 0010-4620 the whole document	1-7

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 September 2004	22/09/2004

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Carciofi, A

Form PCT/ISA/210 (second sheet) (January 2004)

INTERNATIONAL SEARCH REPORT

ational Application No
/IB 03/01013

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 01/57611 A (BCL COMPUTERS INC) 9 August 2001 (2001-08-09) page 5, line 17 - page 6, line 6 page 13, line 9 - line 15 page 15, line 11 - line 23	1-7
A	WO 01/93097 A (NOKIA CORP ; NOKIA INC (US)) 6 December 2001 (2001-12-06) abstract page 11, line 8 - page 14, line 26	1-7
A	GB 2 366 037 A (IBM) 27 February 2002 (2002-02-27) page 7, line 33 - page 10, line 21	1,5-7

INTERNATIONAL SEARCH REPORT

Information on patent family members

ıational Application No
. . ./IB 03/01013

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 0157611 A	09-08-2001	WO 0157611 A2	09-08-2001
WO 0193097 A	06-12-2001	US 6556217 B1	29-04-2003
		AU 5659301 A	11-12-2001
		EP 1316025 A2	04-06-2003
		WO 0193097 A2	06-12-2001
GB 2366037 A	27-02-2002	CN 1310415 A	29-08-2001

Attorney's Docket No. 042933/293426 PATENT
Express Mail Label No. EV 521115696 US

PROVISIONAL APPLICATION FOR PATENT COVER SHEET

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 C.F.R. 1.53(c).

INVENTOR(s)

Name: Steven C. Greer
Residence: 3201 Toler Road
Rowlett, Texas 75089

Name: Philippe Gournay
Residence: 855 rue du mont Brome
Sherbrooke (Quebec)
J1L2V9 Canada

Name: Milan Jelinek
Residence: 841 Montcalm No. 1
Sherbrooke (Quebec)
J1H1J2 Canada

TITLE OF THE INVENTION *(500 characters maximum)*

SYSTEM AND METHOD FOR ADAPTIVE TRANSMISSION OF SILENCE DESCRIPTOR (SID) FRAMES, COMFORT NOISE (CN) PARAMETER MESSAGES OR THE LIKE

CORRESPONDENCE ADDRESS

Andrew T. Spence
Registration No. 45,699
Customer Number 00826

Tel Charlotte Office (704) 444-1000
Fax Charlotte Office (704) 444-1111

ENCLOSED APPLICATION PARTS *(check all that apply)*

- ☒ Specification (Number of Pages 17)
- ☒ Drawing(s) (Number of Sheets 5)
- ☐ Specification sequence listing (if applicable)
  - a. ☐ Submitted as paper copy (Number of Pages)
  - b. ☐ Submitted on CD-Rom or CD-R (2 copies) [If submitted on CD the specification must contain a cross-reference paragraph that incorporates by reference the contents of the CD.]
    - i. ☐ A statement verifying identity of CD copies
- ☐ Application Data Sheet. See 37 CFR 1.76
- ☐ CD(s), if other than Sequence Submission, Number
- ☐ Other (specify)

METHOD OF PAYMENT OF FILING FEES

- ☐ Applicant claims small entity status
- ☒ Check or money order is enclosed to cover the filing fee.
- ☐ The Commissioner is hereby authorized to charge filing fees to Deposit Account No. 16-0605.
- ☒ Please charge Deposit Account No. 16-0605 for any fee deficiency.

PROVISIONAL APPLICATION FILING FEE AMOUNT(s)

Large Entity $200.00; Small Entity $ 100.00

APPLICATION SIZE FEE 0

(If the specification, drawings, and sequence listing exceed 100 sheets of paper, the application size fee due is $250 ($125 for small entity) for each additional 50 sheets or fraction thereof.)

Total Application Fee : $200.00

The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.

- ☒ No.
- ☐ Yes, the name of the U.S. Government agency and the Government contract number are:

Customer No. 00826
ALSTON & BIRD LLP
Bank of America Plaza
101 South Tryon Street, Suite 4000
Charlotte, NC 28280-4000
Tel Charlotte Office (704) 444-1000
Fax Charlotte Office (704) 444-1111

Respectfully submitted,

Andrew T. Spence
Registration No. 45,699

"Express Mail" mailing label number EV 521115696 US
Date of Deposit: June 17, 2005
I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 on the date indicated above and is addressed to:
Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

_____
Andrew T. Spence

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 October 2003 (23.10.2003)

PCT

(10) International Publication Number
WO 03/088035 A2

(51) International Patent Classification[7]: G06F 9/38

(21) International Application Number: PCT/IB03/01013

(22) International Filing Date: 19 March 2003 (19.03.2003)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
02076460.1 15 April 2002 (15.04.2002) EP

(71) Applicant *(for all designated States except US)*: **KONINKLIJKE PHILIPS ELECTRONICS N.V.** [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **MORRIS, Steven** [GB/]; c/o Prof . Holstlaan 6, NL-5656 AA Eindhoven (NL).

(74) Agent: **GROENENDAAL, Antonius, W., M.**; Internationaal Octrooibureau B.V., Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).

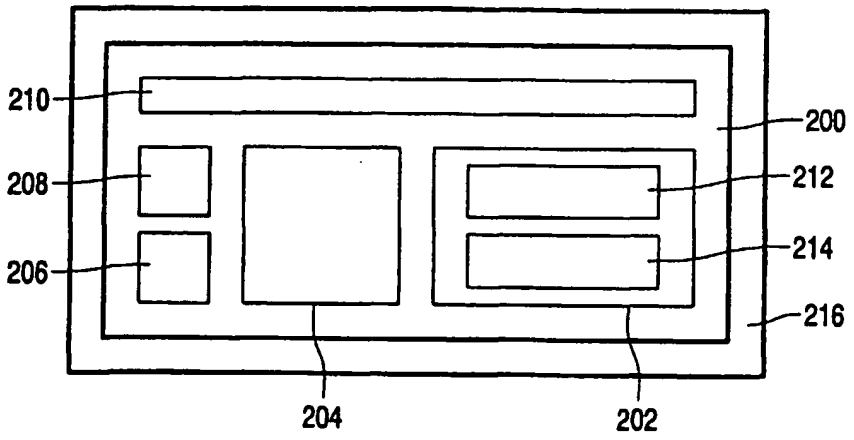
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NI, NO, NZ, OM, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IT, LU, MC, NL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: METHOD, SYSTEM, COMPUTER PROGRAM PRODUCT AND STORAGE DEVICE FOR DISPLAYING A DOCUMENT



(57) Abstract: Handheld display devices have often limited display capabilities. In order to view a full HTML page (200), a method is provided that splits the original page into less complex units (210, 206, 208, 204, 212, 214, 202) so that they can be best displayed on these small devices. The document is parsed into a tree structure and the complexity of the tree is calculated. Depending on the complexity it is decided which part of the document must be displayed onto a new page.

WO 03/088035 A2

Method, System, Computer program product and Storage device for displaying a document

The invention relates to a method of displaying a structured document comprising the steps of:

loading the structured document;

parsing the structured document into a hierarchical structure.

The invention further relates to a system to display a structured document, the system comprising:

loading means conceived to load the structured document;

parsing means conceived to parse the structured document into a hierarchical structure.

The invention further relates to a computer program product comprising program code means designed to perform such method.

The invention further relates to a storage device comprising such computer program product.

An embodiment of such a method is known from US patent 5,987,256. Here a method is described for processing an object specified by an object specifying language such as HTML, JAVA. Also other specifying languages can be used, that rely on relative positioning that requires a rendering program. This rendering program utilizes a minimum set of resources and translates the code for use in a target device that has limited processing resources. These limited processing resources are unsuited for storage and execution of the HTML rendering program, JAVA virtual machine, or other rendering engine. Data concerning such an object is generated by a process that includes first receiving a data set specifying language, translating the first data set into a second data set in an intermediate object language adapted for a second rendering program suitable for rendering by the target device that utilizes actual target display coordinates. The second data set is stored in a machine readable storage device, for later retrieval and execution by the thin client platform. Upon loading, for example an HTML file, into the translating device, information concerning the target device is loaded. The HTML file is then parsed by searching for HTML tags, and based on such tags creating a hierarchical structure. Using the parameters of the target device,

and the hierarchical structure, the method performs HTML rendering based on a hierarchy adapted to the dimensions and palette of the target device. This determines the coordinates of all the graphic objects specified by the HTML code on the screen of the target device. For example, the paragraphs are word wrapped, horizontal rules are placed in particular places, the colors are chosen, and other device specific processes are executed.

It is an object of the invention to provide a method according to the preamble that enables a more flexible adaptation of the contents of a document to display dimensions. To achieve this object the method is characterized in that the method further comprises the steps of:

calculating a complexity of the hierarchical structure;

traversing the hierarchical structure; and

conditionally displaying a part of the structured document depending on the complexity of a traversed part of the hierarchical structure.

By parsing a structured document into a hierarchical structure, for example a Document Object Model tree (DOM) tree, the structured document is subdivided into less complex units. The complexity of this tree structure can then be calculated by calculating the complexity of the units. The complexity of a node of the tree is a measure of the size of the node, preferably including the size of the sub-tree of the node. This size can depend for example on the kind of the unit, like a paragraph or a table, and the amount of document space the unit requires. By using this complexity of a node during traversal of the tree, it is decided on-the-fly if a node and its sub-tree can be comprehensively displayed on a displaying device.

An embodiment of the method according to the invention is described in claim 2. By comparing the complexity of a node with its sub-tree, to a predefined threshold, the parts of the document that can be displayed comprehensively to a user can be determined easily. The threshold can depend on the display dimensions of a display device. The threshold can also depend on user preferences or for example font size used.

An embodiment of the method according to the invention is described in claim 3. By adding a reference to the part of the document that is to be displayed on a separate page, the user does not loose the context of the content of the total document. The user is provided with a common user interface, for example a uniform resource locator (URL), that references the part of the document that is displayed on a separate page.

An embodiment of the method according to the invention is described in claim 4. By dividing the document into elements that are less complex units, the properties of each element can be taken into account to determine the complexity of an element. For example a table element is more complex than a paragraph element, since additional space is required for table borders and cell boundaries. This leads to a higher complexity number for a table than for a paragraph. By taking these complexity numbers into account, it can be decided better if a part of a document can still be displayed on one page by the display device.

It is a further object of the invention to provide a system according to the preamble that enables a more flexible adaptation of the contents of a document to display dimensions. In order to achieve this object, the system is characterized in that the system further comprises:

calculating means conceived to calculate a complexity of the hierarchical structure;

traversing means conceived to traverse the hierarchical structure; and

displaying means conceived to conditionally display a part of the structured document depending on the complexity of a traversed part of the hierarchical structure.

It is a further object of the invention to provide a computer program code means and a storage device that enables a more flexible adaptation of the contents of a document to display dimensions. In order to achieve this object, the program code means is designed to perform the method according to the invention and the storage device comprises the computer program product according to the invention.

The invention will be described by means of embodiments illustrated by the following drawings

Figure 1 illustrates a BBC news site;

Figure 2 illustrates an example of a schematic table layout;

Figure 3 illustrates the main steps of the method according to the invention in a schematic way;

Figure 4 illustrates an example of a partitioning of a table hierarchy comprised within a page;

Figure 5 illustrates the main parts of a device comprising a system according to the invention in a schematic way.

More and more devices are becoming internet-enabled, and the number is only expected to grow in the future. As internet access becomes more ubiquitous, and the devices that provide this access become more mobile, the size of the display that most people use to view internet content will reduce. At present, most internet content is authored to look its best on larger display devices such as computer monitors. Even when displayed on a screen that is relatively large for a mobile device, such as a personal digital assistant (PDA) screen, the usability of content can drop dramatically. A number of factors, such as page complexity, navigational aids and the suitability of content affect the usability of internet content. One of the most common themes is simplicity of design and document structure. This is even more important on a mobile device. For instance, a typical browser running on a PC may have a window size of 800x600 pixels for viewing content. This does not include other screen real estate used by a web browser for menus, toolbars and other features. Even on a high-end mobile device, cost and practicality issues limit the total screen size of 320x240 pixels at present. Mobile phones may even have a display that is only 100 pixels square. Attempting to display a conventional web page that has been authored for a large screen on a small device causes problems to the user because so little of the page is visible on screen at once. Thus, the user loses the context of where they are on the page, and the navigational complexity of the page is increased. This causes problems for web authors wishing to target mobile devices, because the mobile devices have usability requirements far different from conventional desktop PCs.

Current services for mobile devices, such as Wireless Applications Protocol (WAP) or I-mode solve this problem by using markup languages that are subsets of HTML functionality as defined by the World Wide Web Consortium (W3C). In the case of WAP, this is a very different markup language with additional structural features that are used to improve navigation (the "deck of cards' metaphor in WAP). I-mode uses a cut-down version of HTML with much of the functionality removed. In both cases, content must be re-authored or authored in a common format and automatically adapted for use on one or more device types, which can lead to errors, inconsistencies and increased maintenance effort. The overall effect of this is that content is primarily published for one device type.

Tables are often used, by web site designers, to provide control over formatting a Web page that HTML was never intended to provide. Rationales for this can be:

to provide consistent look-and-feel across different web browsers;

to comply with house style rules aimed at printed material rather than web-based material;

to enable stylistic effects that would not be possible otherwise; or

to provide a way of grouping certain elements on a page in a way that fits with a house style.

Figure 1 illustrates the BBC news site http://news.bbc.co.uk. This news site uses eleven tables, nested up to four deep, to maintain their layout. The illustration shows approximately half of the page contents – even on a high-resolution PC display, the user must scroll to see a large portion of the page. The content is roughly three times wider and four times taller than the display on a high-end mobile device. This is a high level of complexity, and is very common amongst web sites.

This level of complexity cannot be displayed easily on a small display device, and for this reason usability is greatly affected. The user both loses context about where they are on the page, and is forced to perform more user interface operations, like clicking, scrolling, etc. to find the information that they want to see. Providing context and reducing the need for user interaction can improve the usability. Techniques such as scaling images and summarizing text are useful aids in usability, but in cases like those illustrated above, the inherent complexity of the document decreases its usability on a device with a small display. A way to improve the usability is to reduce this inherent complexity.

Figure 2 illustrates an example of a schematic table layout. The container table 200 comprises sub-tables 202, 204, 206, 208, and 210. The sub-table 202 comprises sub-sub tables 212 and 214. In order to reduce the complexity of container table 200, a proxy server implements the method according to the invention. A proxy server is a well-known and commonly used mechanism for allowing devices to access internet content. Proxy servers take requests for internet content and pass these requests on to the server that actually contains the content, passing the returned content on to the requesting client. For instance, this is used to provide internet access through firewalls, or to adapt content before it is sent to a client. The proxy server that implements the method according to the invention modifies the contents of for example an HTML document to reduce the complexity of a web page. Documents that adhere to other formats like XML, XHTML, etc. can also be modified to reduce the complexity of the page.

The container table 200 is displayed on a web page 216. By removing, for example, the sub table 202 tables from the main page 216, the complexity of the page 216 is reduced and the page becomes easier to navigate. Reducing the complexity of the page 216 is performed in two main ways:

page breaks are inserted in long pages to reduce the amount of content on each page; and

nested tables may be placed on separate pages, depending on their complexity. The use of tables for formatting enables a web page to be split into coherent sections which can be placed on separate pages, with hyperlinks to those sections instead of the original content. A page with many nested tables can be considered as a tree structure, where each nested table consists of a node in the tree. It is possible to limit the complexity of the contents of a web page by partitioning this tree.

Figure 3 illustrates the main steps of the method according to the invention in a schematic way. Step S300 is an initialization step within which the proxy server receives the document. Within step S302, the proxy server parses the document and creates a parse tree for it. The created parse tree adheres to the Document Object Model (DOM). DOM is a programming interface specification developed by the World Wide Web Consortium. However, the parse tree can also be a less detailed tree that is constructed by a stream-based HTML parser. This stream-based HTML parser searches for the special HTML tags and creates a more simple tree based on these special HTML tags. The stream-based parser, parses the page into its component page elements. These are individual parts of a page that affect the overall structure and formatting of a page, not just of an individual piece of text. The following are considered as separate page elements: paragraphs, tables, lists, preformatted text, images, forms, Java applets.

Within the next step S304, the complexity of each element in the document is calculated. The complexity of each page element is measured as the size of its displayable content, i.e. graphical elements that are actually displayed on the screen, multiplied by a weighing factor to account for the complexity introduced by the page element itself. For example, a table is more complex than a simple paragraph, since extra space is required for table borders and cell boundaries, and so its weighting factor is higher. Some page elements such as lists, forms and tables may contain nested page elements e.g. images or multiple paragraphs within a list entry, and so the complexity of these nested table elements is added to the complexity of the page element that includes them. This complexity value is a property of the document itself, in stead of a property of the display device. Effectively, it is a measure of the size of the document tree, where the "size" of each node may vary with the type of the node. Only a threshold value, as described below, varies with the display size or other external factor. For example, consider the page as illustrated within Figure 4. Then the complexity is measured as follows. First the complexity measure of the list, referred to as *m*

in this example, is taken. Then, an additional complexity factor is added for each of the two list entries, and is referred to as n. Thus, the complexity measure for the list and its structure is

$$(m+2n)$$

This only considers the complexity of the list structure itself. Thus only, the effect of horizontal and vertical spacing required to separate the list from the surrounding text, and to separate each element. It does not consider the complexity of the actual contents of the list. The complexity of the actual contents of the list is calculated separately, and the complexity of the list entries and the list structure are summed to give a total complexity measure for the list. Once the complexity of the list structure itself is calculated, the complexity of the page elements making up each list entry is considered. The first entry consists of two paragraphs. For each paragraph, the complexity is taken to be a constant weighing factor p multiplied by the length in characters of the displayable text in the paragraph. Thus, the complexity of the first list entry as a whole can be considered to be

$$p(para_1+para_2)$$

where $para_1$ and $para_2$ are the length of the first and second paragraphs respectively. The second list entry comprises of one paragraph of text and its complexity is measured as described above. It also comprises an image, whose complexity is measured as a weighting factor i, multiplied by its area a. This gives a measure of complexity for this list element as

$$p(para_3)+ia$$

Thus, the complexity of the entire list can be calculated as

$$(m+2n)+(p(para_1+para_2))+(p(para_3)+ia)$$

The complexity of a table is measured as the sum of the complexity of all cells in the table, multiplied by a weighing factor consisting of a base weighting factor for the table t multiplied by a weighted value for the number of rows (w_{rows}) and a weighted value for the number of columns ($w_{columns}$):

$$tw_{rows}w_{columns}\left(\sum_{cell_0}^{cell_{max}} complexity(cell_n)\right)$$

The value of the weighting factor for rows and columns is constant for each table. Other contributions to this value, like cell spacing, padding and border size, are set as part of the whole element and not on a per-cell basis. Therefore, these contributions are not taken into account for calculating the value of the weighting factor for rows and columns and these weighting factors are calculated once for each table.

The complexity of nested tables is not taken into account when calculating the complexity of a table. Since the purpose of calculating the complexity of a table is to partition the table in to sub-trees based on this complexity measure, the complexity of each node in the tree must not include the complexity of any child nodes since these child nodes may not appear on the same page when the tree is partitioned, and thus will not contribute to its complexity in that situation.

Within step S306 a node of the parse tree is considered such that the parse tree is traversed in a depth-first manner. Within the next step S308, the complexity of the node is added to the current complexity count. This current complexity count is compared to a threshold within step S310. The threshold value depends on a number of, non-limiting properties, like the display resolution, font size and user preferences. If the current complexity count is below the threshold, the node, or page element is written to the current page within step S312. If the current complexity count is greater than the threshold, the method proceeds to step S314. Within step S314, a new page is created and the current complexity count is reset. Within the next step S316, a hyperlink, like a unified resource locator or URL, to the new page is inserted into the current, old, page and the method proceeds to step S308. Now, within step S308, the current page considered is the new page. If the page element is written to a page, the method proceeds to step S306 and considers the next node. When there are no more nodes to traverse the method proceeds to step S320 and stops.

For an HTML table, the method proceeds slightly different. When a table is written to the adapted page, the contents of each cell is written out one cell at a time by traversing the document tree in a depth-first manner. If, in the course of processing that table, a nested table is encountered and is too complex to be placed on the current page, the current page and its complexity count is pushed on to a stack of currently open pages. A new page is created for the nested table, and a hyperlink to it is added to the current cell of the current page. The nested table is then written to the new page. Once the nested table has been completely written, the page is closed, and the old page is popped from the stack, so that the remainder of the original table can be written. This is a recursive operation, since tables may be nested to an arbitrary depth.

In pseudo-code, the method for writing the adapted table is as follows:

```
function write_table(
        paragraphElement table
```

```
    ) {
            complexity_count = complexity_count +
                table.table_complexity
            if (complexity_count > threshold_value) {
                create_new_page
                add_hyperlink_to(new page)
                push(current_page, complexity_count)
                current_page = new_page
            }
            for each cell c in current_table {
                for each paragraph element pe in c {
                    if (pe is a table) {
                        write_table(pe)
                    } else {
                    write_paragraph_element(pe)
                    }
                }
            }
            if (
                (table_is_root_table_on_current_page) and
                not (table_is_root_of_table_hierarchy)
            ) {
                pop(current_page, complexity_count)
            }
    }

    function write_paragraph_element(
            paragraphElement pe
    ) {
            if (pe is list) {
                write_list(pe)
            } else if (pe is paragraph) {
                write_paragraph(pe)
```

```
} else if (pe is table) {
        write_table(pe)
} else
...
}
```

Figure 4 illustrates an example of a partitioning of a table hierarchy comprised within a page 400. The tables are numbered in the order that they would be processed, showing that the software traverses the table hierarchy in a depth-first manner. This is consistent with writing every table as they are encountered in the HTML source. Tables 402 and 404 are both fairly simple, and can be written on to the same page. However, table 408 is too complex to be written at a third level of nesting on the current page 426. The method according to the invention creates a new page 424, and writes table 408 onto that page 424. In doing so, the method encounters table 410, which is simple enough to be written to the same page 424. After completing table 410, and the remaining cells in table 408, the method finishes the current page 424 and returns to the previous page 426 and continues to write table 404. When table 406 is encountered, it is simple enough to fit on the same page as tables 402 and 404. After completing the processing of table 404, the method encounters table 412. This is sufficiently complex to require a new page 428. This process is continued for all other sub-tables in the hierarchy.

The method according to the invention is described with reference to HTML pages. However, the method is not limited to HTML pages, but can be also applied to pages based on other hierarchical oriented languages as defined by the W3C, like, for example, XML, XHTML, RDF etc. without departing from the design principles of the current method.

Figure 5 illustrates the main parts of a device 500 comprising the system 502 according to the current invention in a schematic way. The system 502 comprises computer readable code 506 that is designed to load the HTML document. The system further comprises computer readable code 504 that is designed to parse the HTML document into a document tree structure as previously described. The computer readable code 508 is designed to calculate the complexity of the HTML document whereas the computer readable code 514 is designed to traverse the document tree in a depth-first manor to decide if a page element is to be displayed on the current or on a next, newly created page. The computer readable code 512 is designed to display the current and newly created pages onto the display of the device

500. The computer readable code is comprised within a general purpose memory that communicates with the central processing unit 510 through software bus 516. The device 500 is a personal digital assistant (PDA), but can be any handheld display device like a mobile phone or the like that has limited display capabilities. The device can also be a set-top box or a digital television receiver. The device 500 has a wire-less connection to the internet 522. The document that the device receives is comprised onto a server 520. The document can be accessed by the device through the internet 522. The connections between the server 520 and internet 522 is wire-less. Both connections can also be wired. The previously mentioned computer readable code that is designed to perform the method according to the invention can be downloaded from the internet 522 to the device 500. It can also be downloaded from a computer readable medium like a compact disk 518 that comprises the computer readable code 524 that is designed to perform the method according to the invention. In the latter case, the device 500 comprises an appropriate reading device like a compact disk reader.

CLAIMS:

1. Method of displaying a structured document comprising the steps of:

loading the structured document;

parsing the structured document into a hierarchical structure

characterized in that the method further comprises the steps of:

calculating a complexity of the hierarchical structure;

traversing the hierarchical structure; and

conditionally displaying a part of the structured document depending on the complexity of a traversed part of the hierarchical structure.

2. Method of displaying a structured document comprising according to claim 1, wherein the complexity is compared with a predetermined threshold to determine a first part of the document to be displayed on a first page and a second part of the document to be displayed on a next page.

3. Method of displaying a structured document comprising according to claim 2, the method further comprising adding a reference to the first page to enable navigation to the second part of the document.

4. Method of displaying a structured document according to claim 1, wherein the document comprises elements that contribute to the hierarchical structure and a property of each element is used to calculate the complexity of the hierarchical structure.

5. System to display a structured document, the system comprising:

loading means conceived to load the structured document;

parsing means conceived to parse the structured document into a hierarchical structure

characterized in that the system further comprises:

calculating means conceived to calculate a complexity of the hierarchical structure;

traversing means conceived to traverse the hierarchical structure; and

displaying means conceived to conditionally display a part of the structured document depending on the complexity of a traversed part of the hierarchical structure.

6. Computer program product comprising program code means designed to perform the method according to claim 1.

7. Storage device comprising the computer program product according to claim 6.

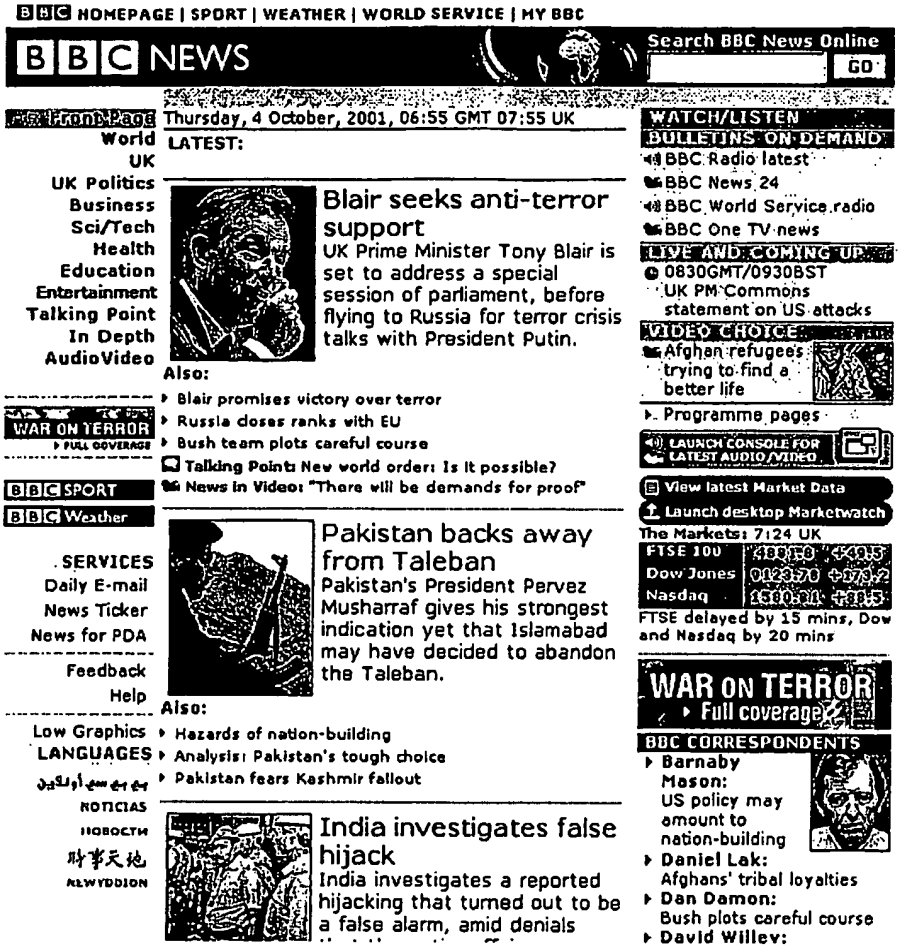

BBC HOMEPAGE | SPORT | WEATHER | WORLD SERVICE | MY BBC
BBC NEWS
Search BBC News Online
GO
Front Page
World
UK
UK Politics
Business
Sci/Tech
Health
Education
Entertainment
Talking Point
In Depth
AudioVideo
WAR ON TERROR
FULL COVERAGE
BBC SPORT
BBC Weather
SERVICES
Daily E-mail
News Ticker
News for PDA
Feedback
Help
Low Graphics
LANGUAGES
Thursday, 4 October, 2001, 06:55 GMT 07:55 UK
LATEST:
Blair seeks anti-terror support
UK Prime Minister Tony Blair is set to address a special session of parliament, before flying to Russia for terror crisis talks with President Putin.
Also:
Blair promises victory over terror
Russia closes ranks with EU
Bush team plots careful course
Talking Point: New world order: Is it possible?
News in Video: "There will be demands for proof"
Pakistan backs away from Taleban
Pakistan's President Pervez Musharraf gives his strongest indication yet that Islamabad may have decided to abandon the Taleban.
Also:
Hazards of nation-building
Analysis: Pakistan's tough choice
Pakistan fears Kashmir fallout
India investigates false hijack
India investigates a reported hijacking that turned out to be a false alarm, amid denials
WATCH/LISTEN
BULLETINS ON DEMAND
BBC Radio latest
BBC News 24
BBC World Service radio
BBC One TV news
LIVE AND COMING UP
0830GMT/0930BST
UK PM Commons statement on US attacks
VIDEO CHOICE
Afghan refugees trying to find a better life
Programme pages
LAUNCH CONSOLE FOR LATEST AUDIO/VIDEO
View latest Market Data
Launch desktop Marketwatch
The Markets: 7:24 UK
FTSE 100
Dow Jones
Nasdaq
FTSE delayed by 15 mins, Dow and Nasdaq by 20 mins
WAR ON TERROR
Full coverage
BBC CORRESPONDENTS
Barnaby Mason: US policy may amount to nation-building
Daniel Lak: Afghans' tribal loyalties
Dan Damon: Bush plots careful course
David Willey:

FIG.1

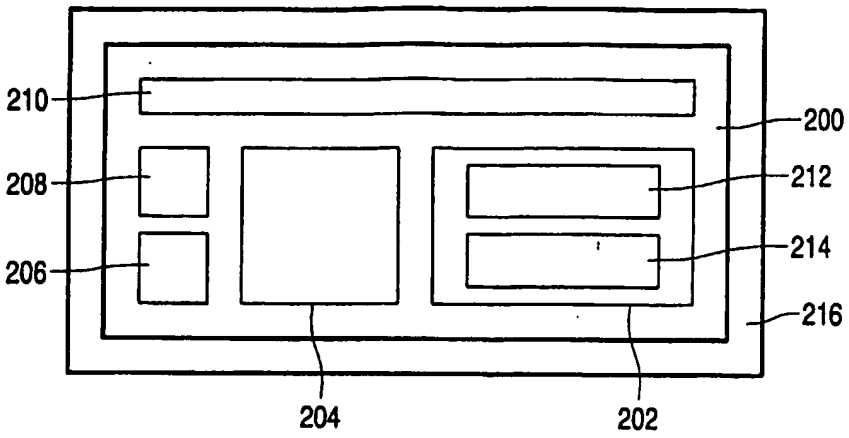

210
208
206
200
212
214
216
204
202

FIG. 2

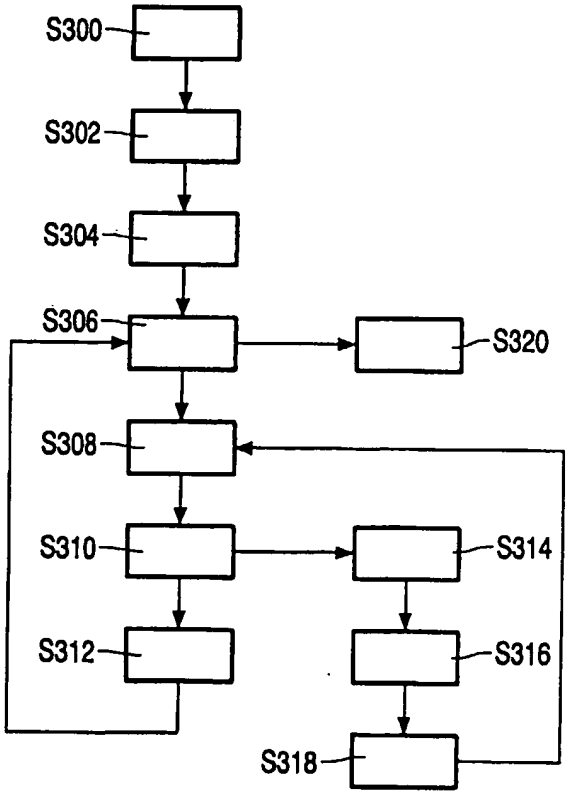

S300
S302
S304
S306
S320
S308
S310
S314
S312
S316
S318

FIG. 3

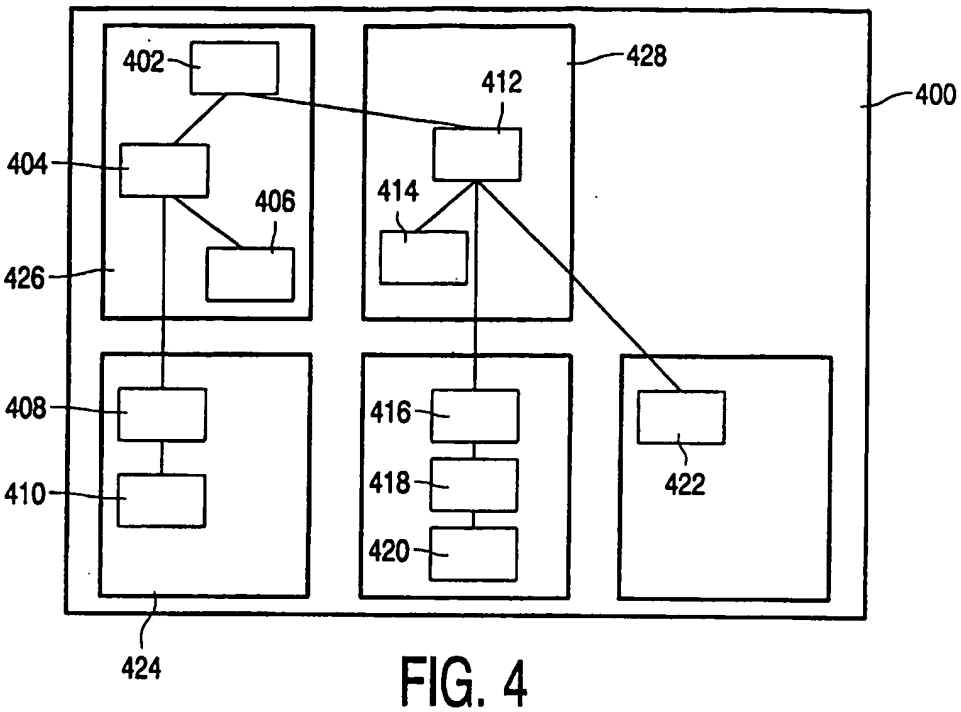

402
428
412
400
404
406
414
426
408
416
418
422
410
420
424

FIG. 4

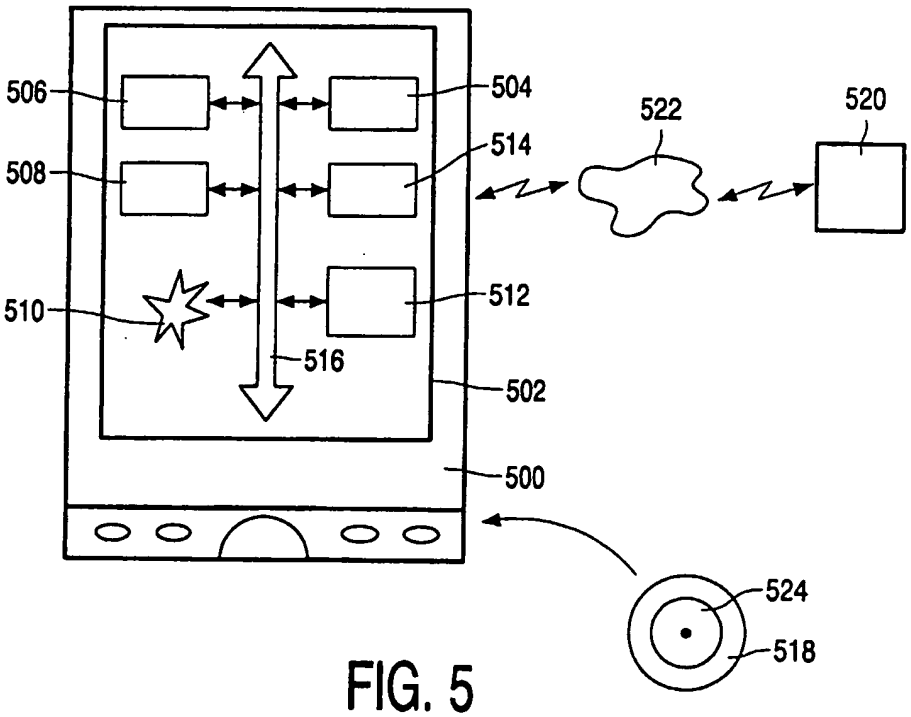

506
504
522
520
514
508
510
512
516
502
500
524
518

FIG. 5